

Control System of SR Source SIBERIA Commissioning and Future Plans

V. Fransev, V. Goluzov, A. Kadnikov, Y. Krylov, S. Kuznetsov, A. Valentinov, Y. Ypinov

Kurchatov Institute, 123182 Moscow, Russia

Abstract

The SIBERIA SR complex, which has been under construction for the past few years, has become operational. The facility has been successfully commissioned using a control system. The paper describes the architecture, the techniques used and the commissioning experience. An implementation of UNIX/MS-Windows compatible software tools, such as LabView, for accelerator control is presented. This paper also reports the hardware-software solutions for the automatic process sequence of injection, accumulation and energy ramping. The hardware approach for the temperature monitoring, timing system, feedback control and turn-by-turn beam diagnostics is presented.

1. INTRODUCTION

The SIBERIA accelerator complex includes an 80 MeV electron linac, a 450 MeV booster storage ring, a 2.5 GeV main storage ring and two transport lines for electron beams. It that was developed by the Budker Institute of Nuclear Physics (Novosibirsk) for the Kurchatov Institute (Moscow)[1]. During the first months of 1995 the main ring was commissioned. The ramping of energy in the main ring is the goal for the next stage.

2. COMMISSIONING RESULTS.

2.1 Hardware

The control system is separated into three layers. A PC network provides the upper level for use in the control room. Four IBM PC 486 computers with 19" color monitors and 8 MB of RAM form the consoles, running MS-Windows for WorkGroups 3.11. Tasks may be interchanged between computers using Network Dynamic Data Exchange (NetDDE). A single PC with a 50 MB Hard Disk forms the interconnection between the consoles and the second real-time layer.

The second layer is composed of eight 24-bit CAMAC-oriented diskless computers, connected to the PC network by 1 Mb/s serial links. They run the real-time multitasking ODOS operating system and are equipped with RS-232 interfaces for local terminals, RAM, network interface, graphic adapter and peripheral crate driver. Each computer controls a specific part of the accelerator (injection, main ring) or a specific system (vacuum, beam monitoring, temperature measurement, etc.)

The third layer consists of 36 CAMAC crates fitted with a total of 385 I/O modules of various kinds. These peripheral crates are placed near the power supplies and the RF and vacuum equipment. They are connected to the control computers by 1 Mb/s serial links on coaxial cables. The distance between crates is 50 to 100 m.

An important feature is the synchronization between the main ring RF, the booster RF and the fast deflectors for injection into the main ring. Multi-channel programmable CAMAC timing generators with a 0.4 ns resolution and a maximum delay of 5.2 μ s are used for frequency matching and selection of the number of bunches. The operator can select the bunches, control the phase of injection and measure the time delay with an accuracy of 1 ns. The RF control includes 16DAC channels for cavity voltage control, feeder current setting and revolution frequency control.

The control computer supplies the control and feedback functions for all except the temperature monitoring system. These are based on an intelligent ADC module, which contains a CPU, RAM and program ROM. It has four relay outputs for the interlock system of the power supplies. The main features of the system are:

- all channels are sampled within a 1 to 5s cycle time
- acquired data are compared with stored warning and alarm levels
- alarm devices and interlocks are activated as required
- a graphic presentation is made of the status and history of selected channels.

A set of 12 movable scintillation probes is used for the first turn diagnostics, the position of each probe being measured to an accuracy of 0.2 mm, using an ADC. The main ring is equipped with 24 horizontal and 24 vertical BPMs, connected to a CAMAC set which includes timer, integrated ADC and amplifier/filter modules. Four pick-ups with special fast 8-bit ADCs can take and store one sample per turn (400 ns) for the first 1000 turns, allowing optimization of the injection into the main ring.

2.2 Software

The most important concept in the software design is that of a "regime" of the storage ring. A regime represents a collection of settings for the control channels; each may be saved in a file and edited by a special program. The ramping process consists of loading a set of regimes for different energies sequentially. During transition from one regime to another the settings in control channels change linearly. If any channel is not present in given regime the set for this channel is not updated during regime loading.

The program PRCS controls the real-time process for changing the facility mode (for example beam acceleration). This program includes following functions: setting the time program, external trap excitation and checking and setting the facility mode in the required number of steps. The PRCS task gives a command to the BANK program for execution. At present there are the following working cycles of the accelerator facility: 1) injection in SIBERIA-1 and beam storage, 2) acceleration to 140 MeV in 50 steps, 3) acceleration to 240 MeV in 100 steps, 4) acceleration to 450 MeV in 100 steps, 5) beam extraction from the SIBERIA-1 and injection to SIBERIA-2, 6) return to injection mode in 300 steps.

The automatic procedure for injection into the main ring SIBERIA-2 was proposed and realized. This approach was based on using the beam-current measurement for feedback. The PRCS program implements injection into SIBERIA-1 until achievement of the setting value of beam current. During this procedure the program compares the set point for the beam current and the measured value from a DCCT. Then it transfers to acceleration in SIBERIA-1 and ramping the energy up to 450 MeV over 30 sec. After that the PRCS automatically generates a sequence of commands for extraction from SIBERIA-1 to the main ring.

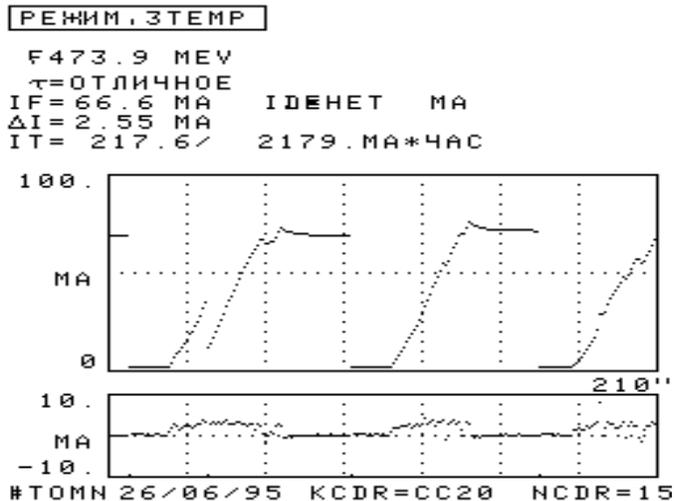


Figure 1. Injection complex automatic regime illustration.

The time of the injection cycle in this case is approximately 60 sec with a SIBERIA-1 beam current of about 80 - 100 mA. In this case the operators and physicist team can control and study the main ring injection without control commands to the injection complex. The operator can change the level of demanded beam current or interrupt the process and come back to manual mode. Figure 1 illustrates this automatic regime of the injection complex. The first graph shows beam current versus time, the full scale of time being 210 sec. The second graph displays addition of beam current each injection cycle into SIBERIA-1. The beam current falls to zero after extraction to SIBERIA-2.

3. NEW PROJECTS

3.1 Local-global digital feedback system for orbit stabilization

The typical beam stability tolerances for modern light sources is about 10% of the beam size and divergence. For SIBERIA-2 this criterion means that we have to position the beam orbit accurate to the 10-15 μm level. Active feedback systems, which detect and counteract undesired beam fluctuations by correcting the closed orbit, are essential to achieve the necessary beam stability. Potential sources driving the electron beam fluctuations are ground vibrations, magnet power supply ripple and thermal drift. A specific feature of the low-emittance light source magnetic lattice is the significant amplification factor for any mechanical vibrations of the magnetic elements due to the strong focusing. In the case of SIBERIA-2 any quadrupole lens motion will transfer to beam motion with the amplification factor (rms) $M_{x,z}=30$. To stabilize the orbit oscillations a fast digital feedback system has been proposed [2]. System realization will include a local digital feedback system for single-point orbit stabilization at the first stage. Then we plan to implement several local systems for beam stabilization at insertion device points. The global digital feedback system will provide orbit stabilization at the 10-15 μm level in the 0.5-100 Hz frequency range.

3.2 LabVIEW to OS-9 software interface

The LabVIEW system is very popular for instrumentation applications. We tried to connect a powerful LabVIEW graphical interface to real time applications. The hardware is based on a standard Ethernet network. The structure of the complex is shown in figure 2.

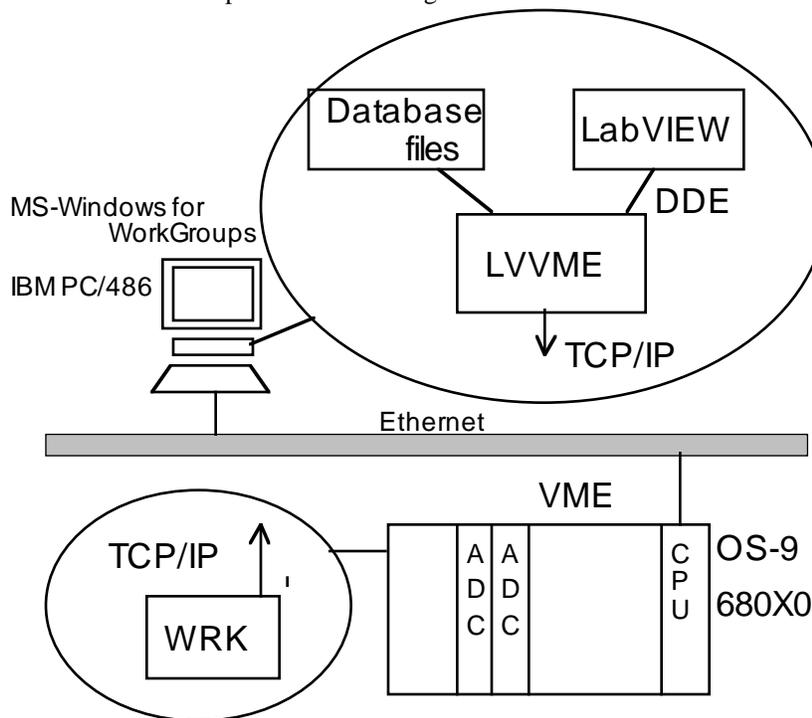


Figure 2. LabVIEW to OS-9 task connection.

The high level software runs on a PC and includes the LabVIEW application and dedicated software server named "LVVME". This server is a MS-Windows application and provides an interface between LabVIEW applications, data base and data-acquisition programs. Intertask communication uses standard MS-Windows Dynamic Data Exchange (DDE) protocol. Connection to the real-time module is based on our "ELib" library. The PC-level software is shown in figure 3.

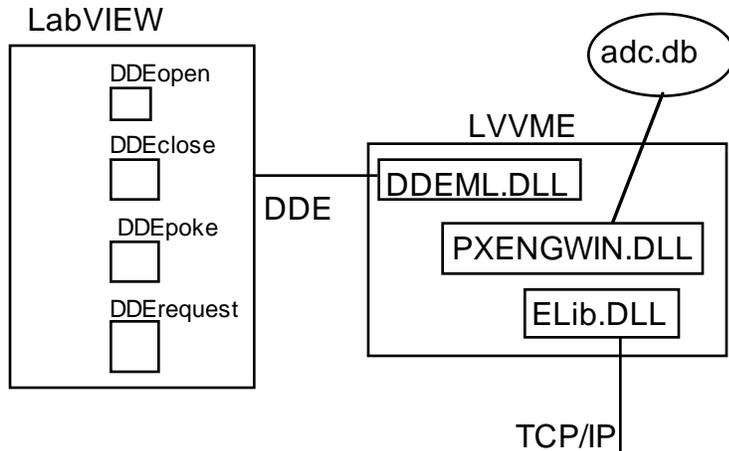


Figure 3. Structure of the LVVME server

For the network interface the special “ELib” library was developed. The library is based on Windows Sockets under MS-Windows and on Internet Support Package under OS-9. The main functions and advantages of the library are:

- asynchronous intertask connection;
- asynchronous data exchange;
- the same program interface under MS-Windows and under OS-9;
- compact (11 functions);
- intertask synchronization;
- TCP/IP implementation;
- one can connect socket and asynchronous event functions.

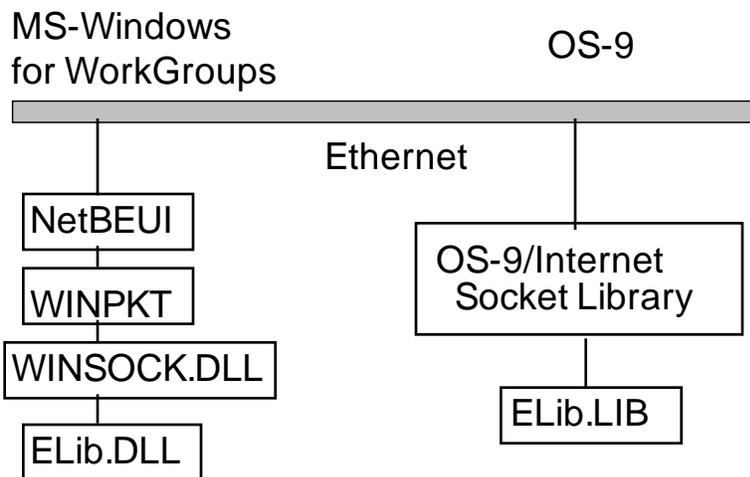


Figure 4. “ELib” up-socket library.

3.3 C++ Libraries

We have developed an object-oriented library for database access and connection between high-level applications and real-time tasks. The database describes the control system of the accelerator complex and consists of a main file and channel families' description files. Each record in the main file includes the name of the channel family, the name of the description file and comments. Hence the number of records is equal to the number of channel families. Family description files describe channels of the same type. For instance, the family of control channels for power supplies, the family of monitoring channels for temperature sensors, the family of measuring channels for digital oscilloscopes, etc. The channel description represents a record in the corresponding file, which contains static information -- names, coefficients, hardware addresses, comments, etc. Channels of different types can have records of different structures. The number of records in a family description file is equal to the number of

channels of this type. Channels have not just to be linked to hardware channels of ADC or DAC, but can also be designed to transmit instructions for a real-time operating system or to produce some mathematical calculations etc. For example, the BeamEnergy channel can be a result of processing the currents in bending magnets. Database access and the mapping of an object oriented library in computer memory has been developed.

Class TDataBase represents the concept of a database as a container of channel families in the form of an array. It also encapsulates methods for database reading from persistent memory and the construction of data members. Similarly, TFamily implements the concept of family as an array of channels. TChannel contains an array of dynamic data (TData) and static data (TStaticData) from the database. TData and TStatData are abstract classes, which include all the necessary methods for information treatment. They implement methods suitable for data of any type. Methods that are different for different data types are presented as pure virtual functions. Classes derived from TData or TStaticData define the implementation of these methods. These are methods of interpretation of database information, treatment of information from the low level control system, execution of requests from console level programs and methods of graphical representation of data.

The library for network and internet exchanges contains classes for communication to control the computer network. TExchange implements the hardware level of information transmission and decoding. TZone executes the link between particular real-time control programs and channels (TChannel). TSocket connects a console level application with the database server program.

Another example of successful implementation of an object-oriented approach is the library for on-line modeling [3]. This library is placed between primitive control channels and complicated 'machine' or 'beam' channels. A class library for on-line modeling is portable and uses the standard MAD input file interface for machine description. The object-oriented conception enables us to realize suitable parent-child class trees for control in "beam" terms.

4. REFERENCES

- [1] V.N. Korchuganov et al., NIM 208 (1983), pp.11-18.
- [2] A.B. Atrakov et al., "Digital Feedback System for Orbit Stabilization at the SIBERIA-2 Light Source", these Proceedings
- [3] S. Kuznetsov "C++ Library for Accelerator Control and On-Line Modeling", these Proceedings.